EXTERNAL REFERENCE SPECIFICATION

For

MP-60 COMPUTER SYSTEMS
Interactive Terminal Subsystem

Prepared By
Application System Development Group
Systems Development Department

# TABLE OF CONTENTS

## 1.0 INTRODUCTION

The Interactive Terminal Subsystem (ITS) extends the full range of MPX/OS features from on-site batch to terminal access. The ITS 2.0 provides operator facilites to the remote terminal user and interactive capabilities to on-site operator. The ITS 2.0 is intended to be used for both software development and applications. Typical applications of the ITS 2.0 for software development are:

    Source program entry.
    Source program maintenance.
    Compilation and assembly.
    Listing inspection.
    User task execution.
    User task debug.
    Documentation generation.

The Interactive Terminal Subsystem supports a wide variety of terminal devices. These may range from simple character mode keyboard/display units to extensive block mode multidevice remote job entry terminals. Terminals may be configured as local hardwired terminals or may be temporarily connected to the system through telephone lines and modems.

The number of interactive users is not limited by the amount of physical memory memory available in the system due to the incorporation of rollin/rollout. As users progress through periods of activity and inactivity, they share the same physical areas of memory.

The ITS 2.0 provides the terminal user with several facilities to monitor and control the MPX/OS. These facilities allow users to:

    Communicate with jobs in the system.
    Control remote and local input and output devices.
    Monitor the system dayfile.
    Control the system input, output, and execution queues.
    Monitor file usage.
    Monitor and control external peripheral devices.

## 2.0 APPLICABLE DOCUMENTS

The following list of Control Data manuals document the MP-60 Emulation, the MPX Operating System and the MPX Product Set:

17329110     MP-32 Computer System MP-60 Emulation Reference
             Manual

17329120     MP-60 Computer System MPX/OS Reference Manual

10992110     MP-60 Computer System MPX/OS Operator's Guide

11110000     MP-60 Computer System Installation Handbook

14291700     MP-60 Computer System Program Command Console
             Reference Manual

14063800     MP-60 Computer System Utility Reference Manual

14062200 /   MP-60 Computer System PRELIB Reference Manual

14061300     MP-60 Computer System COMPASS Reference Manual

14061100     MP-60 Computer System FORTRAN Reference Manual

17328900     MASS/MPSIM Reference Manual

### 3.0  FEATURE DESCRIPTION

### 3.1   PORT CONFIGURATION/USER VALIDATION

### 3.1.1 Abstract

The ability to reconfigure the Interactive Terminal Subsystem at system initialization is provided by the port setup file.

The user validation scheme is based on the Interactive Terminal Subsystem user validation file containing information needed to validate the Username and Password.

### 3.1.2 Description

Immediately upon system loading, a message will appear on the operator console requesting the name of the port setup file. The computer operator must respond to this message before the Interactive Terminal Subsystem will become active.

Each time a terminal user attempts to log on the user validation file is read to validate the username and password and obtain the user's validation mask.

### 3.1.3 Interfaces

The port setup file contains one record for each physical port in the system. The format of each record is as follows:

    did,port,class,mask,p1,p2...pn#

where:

did           represents the device identification (e.g. BCLA01).

port          represents the port number on the device.

class         represents an ITS terminal class as defined for the CLASS command.

mask          represents a string of 0 to 32 hexidecimal
              characters defining the port's validation mask.

*username or null →*

p1,p2...pn    represents a string of port setup parameters as
              defined for the SETUP command.

(AUTO LOGIN ALLOWED?)

#            represents an ETX ($03) character.

Each port has associated with it a validation mask that defines which ITS features may be utilized from that port. The validation mask consists of four words where each set bit indicates an ITS feature that is accessable from that port. The validation mask is defined by the mask parameter in the port setup file left justified zero filled. The correlation of mask bits to ITS features is as follows:

```
word 1    bit   0   - All ITS features.
                1   - Batch Control Facility.
                2   - Device Control Facility.
                3   - Dayfile Management Facility.
                4   - File Control Facility.
                5   - Interactive Communications Facility.
                6   - Operator Control Facility.
              7-9   - TBD.
            10-15   - Security Level.
            16-31   - TBD (to be used by BCF).

word 2    bit  0-15 - TBD (to be used by DCF).
            16-31   - TBD (to be used by DMF).

word 3    bit  0-15 - TBD (to be used by ECF).
            16-31   - TBD (to be used by ICF).

word 4    bit  0-15 - TBD (to be used by OCF).
          bit 16-31 - not used.
```

The port setup parameters are optional and are only needed if the terminals default port setup is to be different than the standard default port setup.

The user validation file contains one record for each user allowed access to the Interactive Terminal Subsystem. The format of each record is as follows:

    userx,passx,mask#

where:

userx        represents a valid username consisting of 1 to 8
             alphanumeric characters.

passx        represents the password associated with that
             username and consists of 0 to 8 alphanumeric
             characters.

3-2            , Default Facility                          CCS-I002-A

mask          represents a string of 0 to 32 hexidecimal
              characters defining the user's validation mask.

#             represents an ETX ($03) character.

Associated with each user name is a validation mask which is
defined the same as the port validation mask. It serves to
define what ITS features that user may have access to.
However, what features a user has access to may vary depending
upon what port the user is logged on to because accessability
of a feature is a logical product of both the port and user
validation masks.

## 3.1.4 Aborts and Recovery

Not applicable.

## 3.1.5 Errors

If the port setup file can not be opened a message is sent to
the console requesting that the computer operator enter the
name of another port setup file.

If the user validation file can not be opened during a user
validation sequence the username is considered invalid.

An error in a port setup file record causes that record to be
ignored which renders that port inoperative.

Errors are not detected in the user validation file.

## 3.1.6 Performance

Not applicable.

## 3.1.7 Installation Parameters

The standard default port setup for all terminals is an
installation parameter.

The contents of the port setup file is dependent upon the
terminal system configuration.

The contents of the user validation file is site dependent.

CCS-I002-A

## 3.2    PORT ESTABLISHMENT

### 3.2.1 Abstract

The establishment of a port consists of connecting the terminal to the Interactive Terminal Subsystem and successfully passing a validation check.

### 3.2.2 Description

After the communications line has been initialized, connection of a terminal to the Interactive Terminal Subsystem is detected by entering any character at the terminal.    ← SHOULD BE ABLE TO SHUT THIS OFF ON A PORT BASIS. (SETUP)

Upon detection of the terminal connection, a banner message is output to the terminal. This message identifies the system and gives the current date and time. The user is then prompted for username and password entry.

After entering a username and password, the entries are validated using the user validation file. This file has been previously prepared by an external job and contains all valid username/password pairs.

If an entry is invalid, the user is so notified and given a second chance to enter the correct information. If the second attempt also results in an invalid entry, the terminal is disconnected and the user must reinitiate the above sequence to gain access to the system.

If both the username and password are valid, the terminal is established as a logical port in the system and the user has access to the Interactive Terminal Subsystem features.

The user may change passwords at any time follwing log on by entering the PASSWORD command. This command replaces the old password on the user validation file with the new password.

The validation check portion of this feature restricts system access to those users supplying a valid username and password pair. Continued system validation requires discretion on the part of the system users to prevent unauthorized system access.

### 3.2.3 Interfaces

The terminal user causes the Interactive Terminal Subsystem to recognize the connection of the terminal by entering any character at the terminal.

A typical port establishment scenario proceeds as follows:

```
Banner Message:          CONTROL DATA        MPX/OS
                         RESIDENT 2A     LIBRARY 2A
                         04/13/79            17:39:24

Username prompt:         ENTER USERNAME
User entry:              JOHNSON
Password prompt:         ENTER PASSWORD
User entry:              8533635
```

At this point the terminal has been established as a port within the Interactive Terminal Subsystem and may utilize system features.

An abreviated form of username and password entry is available as shown below:

```
Banner Message:          CONTROL DATA        MPX/OS
                         RESIDENT 2A     LIBRARY 2A
                         04/13/79            17:39:24

Username prompt:         ENTER USERNAME
User entry:              JOHNSON,8533635
```

Both the username and password may be entered after the username prompt provided the username and password are separated by a comma.

The following scenerio illustrates failure to gain system access:

```
Banner Message:          CONTROL DATA        MPX/OS
                         RESIDENT 2A     LIBRARY 2A
                         04/13/79            17:39:24

Username prompt:         ENTER USERNAME
User entry:              JOHNSSON
Password prompt:         ENTER PASSWORD
User entry:              8533635
                         INVALID USER
Username prompt:         ENTER USERNAME
```

```
User entry:              JONSON
Password prompt:         ENTER PASSWORD
User entry:              8533635
                         LOGOFF AT   17:39:45
                         CONNECT TIME 0.00 SECONDS.
```

The user failed to enter a valid username and password in two attempts and was disconntected from the system. If a reattempt is desired, the user must reestablish connection of the terminal to the system.

All valid username/password combinations must reside in the User Validation File.

The format of the PASSWORD command is:

    )PASSWORD opass,npass

where opass and npass represent the old password and new password respectively. The new password must consist of 0 to 8 alphanumeric characters.


3.2.4 Aborts and Recovery

Any port recovery attempt must be preceeded by port establishment. This feature does not participate in aborts. *— WHAT DOES THIS MEAN?*


3.2.5 Errors

Any data entry errors made during the user validation sequence will produce the message INVALID USER. If the user fails to correctly enter the data on a second attempt, the terminal is disconnected from the system.

If the PASSWORD command is entered incorrectly the message INVALID COMMAND is produced.


3.2.6 Performance

Not applicable.

### 3.2.7 Installation Parameters

Not applicable.

## 3.3    TERMINAL DATA ENTRY/DISPLAY

### 3.3.1 Abstract

The  Interactive  Terminal  Subsystem  provides  a  common
input/output  format  to  tasks  within  the  system  and  the
terminal user.

### 3.3.2 Description

The Interactive Terminal Subsystem provides a central point in
MPX/OS which handles all data transfers to and from terminals.
This provides the  means to standardize terminal  input/output
formats to both the terminal user and tasks within the system.

### 3.3.3 Interfaces

When the Interactive Terminal Subsystem is prepared to receive
data, the terminal user is prompted with a line feed. The line
feed may be transmitted immediately preceeding output data and
therefore may not be apparent to the user. However, the
Interactive Terminal Subsystem will not transmit data to the
terminal until it is prepared to receive data. Therefore, any
transmission of data to the terminal serves as a prompt.

The Interactive Terminal Subsystem provides the  user with two
modes of data entry,  line mode and block mode. In line mode,
the  received data constitutes a single line,  while in block
mode the received data may constitute several lines.

Line mode is the normal mode of data entry. In line mode, data
entry is terminated by a carraige return ($0D). Following the
entry of a line the Interactive Terminal Subsystem performs
some simple editing of the received data to make it correspond
to what appears at the user's terminal.  This includes
processing backspace ($08), forward skip ($15), and cancel
($18) characters. The cancel character causes all previous
characters to be ignored therefore producing the same result
as backspacing to the beginning of the line. The user also has
the ability to select either the  96 or 64 ASCII character set
and parity.

In block mode, data entry is terminated by an ETX ($03)
therefore allowing imbedded carraige returns ($0D) and line
feeds ($0A). Data entered in block mode is considered to be 8
bit bytes and is not interpreted as characters except when

preceeded by the ITS command character. Therefore data received in block mode is not editted. In addition, character set selection is ignored and no parity is allowed.

Checked?

Following data entry and any preliminary editing the first character is examined. If this character is the ITS command character the received data is folded into upper case and processed as an ITS command. The character string immediately following the ITS command character and up to but not including the first non-alphabetic character must be one of the following ITS commands or an error message is produced.

| | | |
|---|---|---|
| ACTIVE | HELP | PASSWORD |
| BATCH | JOB | PORT |
| CLASS | KEEP | QUIT |
| DELAY | LINK | RECOVER |
| DISPOSE | MESSAGE | SETUP |
| FACILITY | PAGE | |

The terminal user need not enter the entire command word but must enter enough to uniquely identify the command. For commands which accept parameters, a space or comma must seperate the command word from the parameters.

If the first character is not the ITS command character the received data is passed to an ITS facility to be processed.

The transmission of output data is independent of input mode.

### 3.3.4 Aborts and Recovery

Not applicable.

### 3.3.5 Errors

Errors made while entering data in line mode may be corrected by using the backspace, forward skip, and cancel characters.

If the terminal user enters more than the maximum number of characters allowed in line mode the message LINE TOO LONG is displayed.

If the terminal user enters more than the maximum number of characters allowed in block mode the message BLOCK TOO LONG is displayed.

CCS-I002-A

### 3.3.6 Performance

Not applicable.

### 3.3.7 Installation Parameters

The ITS command character is an installation parameter but is normally a right bracket ($50).

The input buffer sizes for line and block modes are installation parameters. The normal buffer sizes are 140 characters for line mode and 2000 characters for block mode.

## 3.4   FACILITIES

### 3.4.1 Abstract

Within the Interactive Terminal Subsystem there are several facilities to provide an interface between the terminal user and the system.

### 3.4.2 Description

Each of the following facilities provides the terminal user with a different interface to the system and has its own commands.

BCF - Batch Control Facility
BCF - Device Control Facility
DMF - Dayfile Management Facility
      File Control Facility
JCF - Interactive Communications Facility
OCF - Operator Control Facility

Following log on the terminal user is in communications with the default facility, however the terminal user may switch facilities at any time by entering the FACILITY command. If the terminal user is not validated to communicate with the requested facility the command is considered invalid and an error message is produced.

These facilities process all data received from the terminal which does not constitute an ITS command.

### 3.4.3 Interfaces

The format of the FACILITY command is:

    ]FACILITY facname

where facname represents one of the following mnemonics.

    BCF - Batch Control Facility
    DCF - Device Control Facility
    DMF - Dayfile Management Facility
    FCF - File Control Facility
    ICF - Interactive Communications Facility
    OCF - Operator Control Facility

When the terminal user switches to the Interactive Communications Facility the message PROCESSX or NO CURRENT PROCESS is displayed where PROCESSX represents the name of the current process. When switching to any other facility the message FACX is displayed where FACX represents the facility mnemonic.

### 3.4.4 Aborts and Recovery

Not applicable.

### 3.4.5 Errors

An error in entering the FACILITY command produces the message INVALID COMMAND.

### 3.4.6 Performance

Not applicable.

### 3.4.7 Installation Parameters

The default facility is an installation parameter but is usually ICF.

### 3.5   BATCH CONTROL FACILITY

### 3.5.1 Abstract

The Batch Control Facility (BCF) provides the terminal user with the capability to control and monitor remote and local input, and output devices.

### 3.5.2 Description

The Batch Control Facility allows the terminal user to control files which are currently being read or printed. The Batch Control Facility user must enter enough characters to uniquely identify the particular command. A space or comma must separate the command from the parameters. The parameters are separated by commas and are usually order-dependent. Default values may be selected by entering two consecutive commas.

The valid BCF commands are:

✓ BSP           causes a output file to be backspaced a specified
                number of 1920 character blocks.

✓ END           stops file transfering and the file transfering
                to the device specified is evicted.

  READ          causes a deck to be read from the local or
                terminal card reader and creates a file which is
                catalogued under the specified file. The file is
                then accessable to the interactive user or may be
                submitted for execution.

  REPEAT        causes the file on the specified device to be
                reprinted the specified number of times.

  REWIND        causes an output file to be rewound and returned
                to the output queue.

  RETURN        causes a file being printed to be rewound and
                returned to the output queue with a specified
                priority.

  SUPPRESS      results in the carriage control characters
                contained in the file being printed to be
                disregarded.

### 3.5.3 Interfaces

The format of the BSP command is:

    BSP did,ss

where did is the device identifier of the output device, the default being LP1, and ss is the decimal number of 1920 character logical blocks to be backspaced.

The format of the END command is:

    END did

where did is the device identifier of an input or output device. There is no default device. The END command when used for a line printer, terminates the current file transfer and evicts the file.

The format of the READ command is:

    READ fname,fo,fed,fak,CRn
    READ CRn

where:

fname represents the Name of the catalogued job file.

fo       represents the Owner of the catalogued job file.

fed      represents the Edition of the catalogued job file.  The
         default is 01.

fak      represents the Access Key of  the catalogued job file.
          The default is an Access Key equal to the Owner.

CRn      is the number of the Card  Reader from which the file is
         to be read. The default is CR1.

The first form of  the READ command is  used  to transfer card images into the specified file for later processing, while the second form is used to create a scratch input file to  receive card images and and be submitted for execution.

The format of the REPEAT command is:

    REPEAT did,m

where did is the device identifier of the output device, and m is a decimal number which specifies the number of additional copies desired. If m is omitted, 1 is assumed.

The format of the REWIND command is:

    REWIND did

where did is the device identifier of an output device, the default being LP1. The file on the device specified is rewound and the device turned logically OFF. The device must be turned logically ON before output will continue.

The format of the RETURN command is:

    RETURN did,p

where did is the device identifier of an output device, the default being LP1, and p is the priority. The default priority is the existing one, but the priority may be a decimal value from 0 to 65535.

Printing stops and the output file is rewound and returned to the output queue with priority p. The residual repeat count for that file is saved and printing resumes after command completion.

The format of the SUPPRESS command is:

    SUPPRESS LPn

where n is the logical device number of the printer. The remainder of the file not already printed is single spaced.


3.5.4 Aborts and Recovery

Not applicable.


3.5.5 Errors

If a BCF command is not recognized or the format is incorrect, the command is ignored and the message INVALID COMMAND is displayed.

If the user enters the READ command while cards are reading, card reading continues and the message INVALID COMMAND is displayed.

### 3.5.6 Performance

Not applicable.

### 3.5.7 Installation Parameters

Not applicable.

## 3.5    DEVICE CONTROL FACILITY

### 3.5.1 Abstract

The Device Control Facility (DCF) provides the terminal user with the capability to control external peripheral devices.

### 3.5.2 Description

The Device Control Facility allows the terminal user to control the status of certain peripheral devices. The Device Control Facility user need not enter the entire command, but must enter enough to uniquely identify the entire command. For commands which accept parameters, a space or comma must separate the command word from the parameters, parameters are separated by commas, and except for the DEFINE command, parameters are order-dependent. Default values can be selected by entering two consecutive commas.

The valid DCF commands are:

CONTINUE        restarts file transmissions which have stopped because a message was sent to the CRT display or as a result of a command. Interrupted file transmissions are automatically restarted after any of the DCF commands or after successful completion of a Batch Control Facility (BCF) command.

DEFINE          specifies certain attributes of an output device. If the device is already configured, DEFINE modifies the attributes of the device. If the device is not configured, DEFINE configures the device with the specified attributes.

GO              causes operations suspended temporarily by the WAIT command to be resumed.

LIST            lists the configuration and status of devices currently defined in the system.

OFF             turns the specified equipment logically OFF, negating the effect of the ON command.

ON              turns the peripheral device specified logically ON, negating the effect of the OFF command or the Batch Control Facility command REWIND.

WAIT causes file transfering to be suspended until a GO command is received.

3.6.3 Interfaces

The format of the CONTINUE command is:

    CONTINUE

The format of the DEFINE command is:

    DEFINE did,p1,p2,p3

where did is the output device identifier, default being LP1. p1,p2,p3 represent a string of order-independent parameters which define nondefault characteristics.

Possible parameters are:

    IP    - Impact Printer.
    NIP   - NonImpact Printer.
    LW=lw - Line Width of the printer. lw can be 80 - 136.
            The default width depends upon the terminal type.

If a user enters a DEFINE command with only the did parameter, the device is returned to a default status. If the user enters a DEFINE command for a device already configured, only those attributes specified are altered.

For terminals with additional devices, the user needs a DEFINE command to configure each additional device.

The format of the GO command is:

    GO did

where did is the device identifier of an input or output device or ALL, the default being LP1. Using the ALL parameter causes reading and printing on all devices temporarily suspended by a WAIT command to resume. A WAIT/GO sequence has no effect on the file contents, if WAIT stops file processing, the file continues from the point where it stopped. If no file activity is in progress when WAIT is entered, GO with an appropriate parameter ensures that printing is initiated as soon as an output file becomes available or when card input is available from the card reader.

The format of the LIST command is:

    LIST

The output created by this command is depicted in Appendix A.

The format of the OFF command is:

    OFF did

where did is the device identifier of an input or output device or ALL, the default being LP1. Using the ALL parameter turns all peripheral equipment logically OFF. Data transfers cannot occur on devices logically OFF. If a device is turned OFF while a data is being transfered, processing continues until the end of file is reached, and no subsequent files are transfered. A device is automatically turned OFF by the Batch Control Facility command REWIND.

The format of the ON command is:

    ON did

where did is the device identifier of an input/output device or ALL, the default being LP1. Using the ALL parameter turns all peripheral equipment logically ON.

The format of the WAIT command is:

    WAIT did

where did is the device identifier of an input or output device or ALL, the default being LP1. Using the ALL parameter causes reading and printing to be suspended at all devices. The transfers can be restarted the GO command.


3.6.4 Aborts and Recovery

Not applicable.

### 3.6.5 Errors

If a DCF command is not recognized or the format is incorrect, the command is ignored and the message INVALID COMMAND is produced. The command should then be corrected and retransmitted.

### 3.6.6 Performance

Not applicable.

### 3.6.7 Installation Parameters

The default set of configured peripheral devices is an installation parameter.

## 3.7 DAYFILE MANAGEMENT FACILITY

### 3.7.1 Abstract

The Dayfile Management Facility (DMF) provides the terminal user with the capability to monitor the system dayfile.

### 3.7.2 Description

The terminal user enters the Dayfile Management Facility to display the system dayfile, selectively dump the system dayfile, and reset the system dayfile. Upon entering the Dayfile Management Facility, system dayfiles messages will being appearing on the terminal as they are processed. The following paragraphs describe the valid Dayfile Management Facility commands.

The TYPE command allows the terminal user to specify a selection criteria. The available selection criteria include:

    JOB Name
    TASK Name
    JOB Sequence number.
    Entry TYPE code

The DUMP command allows the terminal user to selectively dump the system dayfile onto a file. The file may then be scanned and display using the Interactive Communications Facility.

### 3.7.3 Interfaces

The format of a system dayfile entry is as follows:

```
                                                                   1
                     1      1   2       2   2   3     3            3
    1 2         9   1      8   0       7   9   2     5            4
    +-+---------+-+---------+-+---------+-+----+--+------  ------+
    ! ! time    !.!jobname !,!taskname!.!code!, !   message    !
    +-+---------+-+---------+-+---------+-+----+--+------  ------+
```

| character position | Definition |
|---|---|
| 1 | Blank. |
| 2-9 | time, hh:mm:ss |
| 10 | Period. |
| 11-18 | jobname, job identification. |
| 19 | Comma. |
| 20-27 | taskname, task identification. |
| 28 | Period. |
| 29-32 | code, defined as follows:<br>29     Message identification code.<br>30     Event code.<br>31-31 Action code. |
| 33-34 | Comma and Blank. |
| 35-134 | message, dayfile message. |

The format of the TYPE command is as follows:

```
TYPE,JN=jobname,TN=taskname,SQ=seq#,ID=code,EV=code,AC=code
TYPE,ALL
```

where:

jobname    The job name selection criteria.  Only entries with
           the job names matching will be displayed/dumped.

taskname   The task name selection criteria. Only entries with
           the task names matching will be displayed/dumped.

seq#       The unique sequence number generated by the system.

code       The    TYPE    code    divided    into    the    fields:
           Idenification, Event, and Action.

ALL    _   resets the selection criteria to all entries.

The Format of the DUMP command is as follows:

    DUMP,start,stop,ALL

where:

start       is the start time.

stop        is the stop time.

ALL         is a flag to disregard the the TYPE selection
            criteria.


### 3.7.4 Aborts and Recovery

This feature does not participate in any aborts or recovery.


### 3.7.5 Errors

If a DMF command is not recognized or the format is incorrect,
the command is ignored and the message INVALID COMMAND is
produced. The command should then be corrected and
retransmitted.


### 3.7.6 Performance

Accessing the system dayfile will not degrade the real time
capabilities of the MPX Operating System. In order to maintain
the real time capabilities, this feature may not maintain an
accurate dayfile message entry display at the terminal, i.e.
dayfile entries may not be displayed due to dayfile messages
entry rate.


### 3.7.7 Installation Parameters

Not applicable.

### 3.8    FILE CONTROL FACILITY

### 3.8.1 Abstract

The  File Control Facility (FCF)  provides the  terminal  user
with the capability to monitor MPX/OS File Management.

### 3.8.2 Description

The File Control  Facility allows  a  terminal user to display
status  information  about  mass storage  files.  The  command
provided   to accomplish this  is   the LIST   command,  which
displays information about files under a particular owner name
on a particular disk pack,  or lists the device identifiers of
the packs currently mounted.

### 3.8.3 Interfaces

The format of the LIST command is:

```
    LIST O,owner,pack
    LIST A,owner,pack
    LIST D
```

where  O  selects a display  of  all  files currently open,  A
selects a  display  of all allocated files,  and D  selects  a
display of device ID's  for  all disk packs currently mounted.
The owner parameter represents a four character owner name and
is optional. If an owner name is specified only files with the
designated owner name are listed. The did parameter represents
a disk pack device identification and is optional. If a device
identification is specified  only files residing on that  pack
are listed.

Examples of the possible displays are provided in Appendix A.

### 3.8.4 Aborts and Recovery

Not applicable.

### 3.8.5 Errors

If the LIST command is entered incorrectly the command is
ignored and the message INVALID COMMAND is produced.

If an invalid owner name or device identifier is specified,
the message NULL LIST is produced.

### 3.8.6 Performance

Not applicable.

### 3.8.7 Installation Parameters

Not applicable.

## 3.9 INTERACTIVE COMMUNICATIONS FACILITY

### 3.9.1 Abstract

The Interactive Communications Facility (ICF) provides the terminal user with the capability to communicate with jobs in the system.

### 3.9.2 Description

The Interactive Communications Facility does not process any commands. All data received by the ICF is passed to the current process provided there exists a current process and the current process has a read request pending. When a read request is received from the current process the Interactive Communications Facility outputs a prompt. The prompt may be the default ICF prompt or a prompt defined by the process.

*PROMPT BEFORE REQ RCD PLEASE!*

When in communications with ICF write requests from the current process are output to the terminal.

### 3.9.3 Interfaces

There is no command interface between the terminal user and the ICF. The interface to the terminal user consists of data received from the terminal that is not an ITS command and data output to the terminal because of a write request from the current process. The interface between the ICF and the current process is described in Section 3.17 Task/Terminal Communications.

Control of which process is the current process is provided by the LINK and KEEP ITS commands.

### 3.9.4 Aborts and Recovery

Not applicable.

### 3.9.5 Errors

If data is received and the terminal user has no current process the message NO CURRENT PROCESS is produced.

If data is received and the current process has no read request pending the message PROCESSX BUSY is produced where PROCESSX represents the name of the current process.

### 3.9.6 Performance

Not applicable.

### 3.9.7 Installation Parameters

The default ICF prompt is an installation parameter but is usually a question mark and back space ( $3F & $08 ).

## 3.10   OPERATOR CONTROL FACILITY

### 3.10.1 Abstract

The Operator Control Facility (OCF) provides the terminal user with the capability to monitor and alter the MPX/OS input, output, and execution queues, display and alter memory, and respond to operator messages.

### 3.10.2 Description

The Operator Control Facility allows terminal users to monitor system activities. The Operator Control Facility user need not enter the entire command word, but must enter enough to uniquely identify the command. A space or a comma must separate the command word from the parameters.

The valid OCF commands are:

ABORT          causes an executing job to be terminated.

EVICT          removes a file from the input or output queue.

IDLE           causes execution of a task  to be suspended until a RUN command is received.

LIST           displays the current contents of the input, output, execution and operator message queues.

MEMORY         lists and/or alters the current contents of memory in hexadecimal format.

PRIORITY       changes the priority of a file in the input or output queue.

RUN            resumes execution of a task that was suspended by the IDLE command.

WATCH          displays the status of jobs and tasks in execution.

3.10.3 Interfaces

The format of the ABORT command is:

    ABORT jobid

where jobid is the job identifier.

The format of the EVICT command is:

    EVICT jobid,q

where jobid is the job identifier and q is the first initial
of the queue in which the file currently resides. Valid q
parameters are:

    I    Input queue
    O    Print queue

If q is omitted, all files with the job identifier jobid are
elliminated from the input, and output queues.

The format of the IDLE command is:

    IDLE jobid,taskid

where jobid is the job identifier and taskid is the task
identifier of the task within the job that is to be idled.

The format of the LIST command is:

    LIST q

where q is the first initial of the queue to be displayed or
A. The valid q parameters are:

    I    Input queue
    O    Output queue

The format of the MEMORY command is:

    MEMORY S,locn,v1,v2,v3,...
    MEMORY D,locn,n
    MEMORY +m,n
    MEMORY -m,n

where S designates memory is to be altered by storing
v1,v2,v3...vn into memory starting at location locn. The S
parameter causes all altered memory to be displayed up to a
sreen full. The D parameter causes memory to be displayed
starting with location locn and displaying n words. The +m and
-m parameters cause memory to be displayed starting with a
location relative to the previous location displayed. Locn may
be an arithmetic expression with values in Hexadecimal. The n
is the number of locations to display, while m is the
hexidecimal number to be added to the last starting address.
The values to be stored into memory v1,v2,v3...vn must be in
hexidecimal but may be preceeded by a decimal repeat count.
The value 10*0 would result in 0 being stored in 10 successive
locations in memory.

The format of the PRIORITY command is:

    PRIORITY jobid,q,p

where jobid is the job identifier, q is the first initial of
the queue in which the file currently resides, and p is the
new priority to be assigned to the file.

Valid q parameters are:

    I    Input queue
    O    Output queue

The format of the RUN command is:

    RUN jobid,taskid

where jobid is the job identifier and taskid is the task
identifier.

The format for the WATCH command is:

    WATCH
    WATCH jobid
    WATCH jobid,taskid

where jobid is a job identifier and taskid is a task
identifier. If no job identifier is specified status
information is displayed for all jobs in execution. If a job
identifier is specified but no task identifier status
information is displayed for all tasks within the job. If a
task identifier is specified status information is displayed
for that task only. Once the command has been accepted as

valid, the display will be updated each time the job progresses to a new control card until a new OCF command is entered.

### 3.10.4 Aborts and Recovery

Not applicable.

### 3.10.5 Errors

If a OCF command is not recognized or the format is incorrect, the command is ignored and the message INVALID COMMAND is produced. The command should then be corrected and resubmitted.

### 3.10.6 Performance

Not applicable.

### 3.10.7 Installation Parameters

Not applicable.

## 3.11  JOB SUBMISSION

### 3.11.1 Abstract

The terminal user  must submit cataloged jobs to the system in
order to perform work.  At the time of submission, disposition
of  output files  may be  specified.  Files  disposed to  mass
storage may be retained on the system for future use or may be
printed or otherwise disposed of.

The command which submits a cataloged job  file to  the system
determines whether the job may be placed into the system input
queue and whether the job has an  originating port.  The first
job control statement of the  cataloged job determines whether
the job will execute as a memory resident job or is subject to
rollin/rollout.

### 3.11.2 Description

All work to be  done by the terminal  user must begin with the
submission of a cataloged job to the system. The cataloged job
may be created  using the *SAVE  control  card from the system
standard input device, or it may be created by a job executing
within the system. The terminal user can create cataloged jobs
with the text editor or other utility programs.

Two commands are  provided to  perform job submission from the
terminal:  the  BATCH command  and  the  JOB  command.  These
commands  require  input file  identification  and  allow  the
specification of standard output file and standard
 file dispositions.

If an output or file is created with mass storage disposition,
a message is produced giving the name by which the file may be
retrieved.  These files are not accessible through  the normal
file  management  job control  statements  because  they  have
special file identifications.

The  BATCH  command is intended  to  submit  jobs which do not
interact  with  the  terminal.  The Batch command  attempts to
place the job in  the named file into immediate execution.  If
the  job cannot be placed into immediate  execution then the
file is added to the system input queue. If the file cannot be
placed into the queue,  a diagnostic message is produced which
indicates the difficulty.  A job submitted  using  the BATCH
command does not have the originating port information.

The JOB command is intended to submit jobs which will interact with the terminal. The JOB command attempts to place the job contained in the named file to be placed into immediate execution if the resources required by the job's SCHED cards are available. If the job cannot be placed into immediate execution, or if an submission error occurs then a diagnostic message is produced indicating the difficulty. A job submitted using the JOB command has the originating port information.

The first statement of a cataloged job file must be one of the following:  *JOB, *RJOB, *IJOB, or *IRJOB.  The *JOB and *RJOB statements are described in the MPX/OS Reference Manual  and they define a normal memory resident job. The *IJOB and *IRJOB statements are the non-memory resident equivalent statements.

The *IJOB and *IRJOB statements differ from the *JOB and *RJOB statements as follows:

1. Jobs beginning  with *IJOB and *IRJOB  utilize memory pages from the rollin  pool for execution and their initiation is not delayed due to unavailable memory.

2. Jobs beginning with *IJOB and *IRJOB are either placed into immediate   execution   (resources   available)   or   their submission is rejected (resources unavailable).  These jobs are never placed into the system input queue.

3. Jobs  beginning with *IJOB and  *IRJOB cannot be  submitted from the system standard input device.

The  JOB  command  is  intended  to  submit  a  cataloged  job beginning with  a *IJOB  or  *IRJOB  statement,  and the BATCH command is intended to submit a cataloged job beginning with a *JOB  or *RJOB statement;  however,  these conventions are not enforced.

### 3.11.3 Interfaces

The formats of the BATCH and JOB commands are:

    ]BATCH fname,fo,fed,fak,printdisp,[~~~~~~~~~~~]
    ]JOB fname,fo,fed,fak,printdisp,[~~~~~~~~~~]

where:

fname       represents the name of the cataloged job file.

fo          represents the owner of the cataloged job file.

fed         represents the edition of the cataloged job file.

fak         represents the access key of the cataloged job file.

printdisp represents the disposition code for standard output.


The   file   disposition   codes   must   be   legal   MPX/OS   file
disposition codes.

If the mass storage disposition (MS) is selected, the message

     OUTPUT:  xxMS
  or  PUNCH:   xxMS

is produced as appropriate,  where xxMS is a unique identifier
of the file and may  be used  to subsequently  access the file
with the DISPOSE command.

The default fields available with the  BATCH and  JOB commands
differ when the fo  field is specified. When the fo field  is
specified the fed defaults to 01 while the  fak field defaults
to the value specified in the fo field. The default fields for
the BATCH and JOB command are as follows:

     ]BATCH BATCH-DEFAULT,SAVE,00, ,MS,SC
     ]JOB JOB-DEFAULT,SAVE,00, ,SC,SC

Examples of the default field usage is as follows:

     ]BATCH

     Results  in  the  file   BATCH-DEFAULT,SAVE,00,   ,   being
     submitted with the output  disposition of  mass storage and PUNCH
     disposition of scratch.

     ]BATCH MYJOB,,,,PR

     Results in the file MYJOB,SAVE,00,  ,  being submitted with
     the output disposition of the printter  and  disposition of
     scratch.

     ]BATCH MYJOB,FOR

     Results in the  file  MYJOB,FOR,01,FOR being submitted with
     the output disposition  of the printer and  disposition  of

scratch.

If a job submitted by the BATCH command is placed in the system input queue the following messages will be produced:

    JOB IN INPUT QUEUE.
    XXX NOT AVAILABLE.

where XXX represents the standard MPX/OS hardware type of the unavailable resource.

The formats of the *IJOB and *IRJOB job control statements are

    *IJOB(ID= ,AC= )
    *IRJOB(ID= ,AC= )

where the parameters are the same as those for the standard *JOB job control statement.


3.11.4 Aborts and Recovery

Not applicable.


3.11.5 Errors

If an error is made entering the BATCH or JOB command the message INVALID COMMAND is produced.

If the file specified in a BATCH or JOB statement cannot be successfully submitted to the system, the message JOB NOT INITIATED is produced followed by the appropriate message from the following list indicating why the submit failed.

    FILE MANAGER ERROR XX.
    FILE FORMAT ERROR.
    JOB STATEMENT ERROR.
    SCHED STATEMENT ERROR.
    INPUT QUEUE FULL.
    NO TABLE SPACE AVAILABLE.

### 3.11.6 Performance

Not applicable.

### 3.11.7 Installation Parameters

Not applicable.

### 3.12  FILE DISPOSITION

### 3.12.1 Abstract

The DISPOSE command may be used to print or otherwise dispose of mass storage files.

### 3.12.2 Description

The DISPOSE command allows a terminal user to route a file to the system output queue or release a file which is no longer needed. The file being disposed may either be a catalogued file or a file created by BATCH or JOB commands whose disposition was to mass storage. In either case the file cannot be disposed if it is open, and once a file has been disposed it no longer exists.

### 3.12.3 Interfaces

The format of the DISPOSE command is:

```
]DISPOSE disp,fid
         or
]DISPOSE disp,fname,fo,fed,fak
```

where:

disp        represents a an MPX/OS file disposition code.

fid         represents the identifier of the output file as received from the BATCH or JOB command.

fname       represents the name of the catalogued job file.

fo          represents the owner of the catalogued job file.

fed         represents the edition of the catalogued job file.

fak         represents the access key of the catalogued job file.

For a file to be released or routed to an output device it must not be open.

### 3.12.4 Aborts and Recovery

Not applicable.

### 3.12.5 Errors

Errors encountered by the execution of DISPOSE commands result
in the appropriate message from the following list:

    FILE MANAGER ERROR XX.
    UNKNOWN DISPOSITION.
    QUEUE FULL.

### 3.12.6 Performance

Not applicable.

### 3.12.7 Installation Parameters

Not applicable.

### 3.13  PORT STATUS AND CONTROL

### 3.13.1 Abstract

The Interactive Terminal Subsystem provides the terminal user with several commands to control port input and output and monitor port status.

### 3.13.2 Description

Each port established in the MPX/OS system has associated with it various status and parameter information. The following commands allow the user to display and modify this information.

The PORT command produces a display of the user's port number, username, log on time, terminal class, facility, and connected processes.

The ACTIVE command produces an abbreviated display of this information for all active ports in the system.

At the time of port establishment, the terminal class and default port setup is established as defined for that terminal in the port setup file.

The user may declare the terminal to be of a different terminal class at any time by entering the CLASS command.

The user may display or change the port setup information by entering the SETUP command. The port setup is used to control the mode of data transmission to and from the terminal in those implementations which provide variable transmission mode selection.

The user may request a time delay between outputs by entering the DELAY command. This is intended to allow time for the carraige return on terminals which produce a hard copy.

The PAGE command allows the user to define the terminal display size and enables paged output. When paged output is enabled consecutive outputs will be halted when the terminal display is full and will continue when the user enters a carraige return (line mode) or ETX (block mode).

*(ADD)*

TIME: 15:32:46   DATE: 12/25/79

### 3.13.3 Interfaces

The format of the PORT command is:

    ]PORT

Entry of the PORT command produces a display similar to the following:

    PORT: 6     USERNAME: JOHNSON      LOG ON: 14:37:46

    CLASS:      7
    FACILITY:   ICF
    PROCESSES:  EDITOR      *PCC

where EDITOR and PCC are connected processes. The asterisk (*) denotes the current process.

The format of the ACTIVE command is:

    ]ACTIVE

Entry of the ACTIVE command produces a display similar to the following:

    PT   USER       LOG ON    FAC PROCESS
    1 JONES      14:32:47   ICF EDITOR
    5 SMITH      13:23:12   OCF
    6 JOHNSON    14:10:59   DCF PCC

where ports 1, 5, and 6 are the only active ports.

The format of the CLASS command is:

    ]CLASS n

where n represents one of the following ITS terminal classes.

        1     M33, M35, M37, or M38 TTY
        2     CDC 713-10
        3     reserved
        4     IBM 2741
        5     M40 TTY
        6     Hazeltine 2000
        7     CDC 751-1
        8     Tektronix 4000
        9     HASP Protocol
        10    200 User Terminal

```
11     CDC 214
12     CDC 711-10
13     CDC 714
14     CDC 731-12, 732-12
15     CDC 734
```

The format of the SETUP command is:

    )SETUP p1,p2...pn

where p1,p2...pn represents a string of parameters from the
following list.

```
HALF - half duplex          110  - 110 baud
FULL - full duplex          300  - 300 baud
ODD  - odd parity           600  - 600 baud
EVEN - even parity          1200 - 1200 baud
NONE - no parity            2400 - 2400 baud
LINE - line mode            4800 - 4800 baud
BLK  - block mode           9600 - 9600 baud
96   - 96 character set      19.2 - 19200 baud
64   - 64 character set
```

If two or more parameters conflict, the last parameter is
used. If an invalid parameter is entered the entire command is
considered invalid and no changes are made. If no parameters
are present, the SETUP command produces a display of the
current port setup information similar to the following:

```
MODE:   LINE
CHAR:   64
BAUD:   9600
DUPLEX: FULL
PARITY: EVEN
```

The actual transmission mode used with a terminal is subject
to individual site implementation, and the setup parameters
may not be used in all implementations.

The format of the DELAY command is:

    )DELAY n

where n specifies the delay time in hundreths of a second and
is a value between 0 and 99.

*WHAT IS BEING DELAYED?*

The format of the PAGE command is:

    ]PAGE n,m

where n is the number of lines in a page and m is the page width in characters. The maximum number of lines per page is 63 and the page width must be between 1 and 136. Paging is disabled by setting the number of lines per page to 0.


### 3.13.4 Aborts and Recovery

Not applicable.


### 3.13.5 Errors

If errors are made entering these commands the message INVALID COMMAND is produced.


### 3.13.6 Performance

Not applicable.


### 3.13.7 Installation Parameters

Port setup parameters must be configured to agree with the actual site implementation.

### 3.14   INTERPORT COMMUNICATIONS

#### 3.14.1 Abstract

The Interactive Terminal Subsystem allows terminal users to communicate with other terminal users connected to the system.

#### 3.14.2 Description

A terminal user may send messages to other ports and may also receive messages from other ports. The user can utilize the ACTIVE command to determine which ports are active and which user is logged on to each port. Sending messages to an inactive port causes an error message to be produced.

#### 3.14.3 Interfaces

PLEASE ALLOW
/ AS
SEPARATOR

The format of the MESSAGE command is:

    ]MESSAGE n text of message

where n represents the number of the port the message is to be sent to or "ALL" if the message is to be sent to all active terminals.

The format of a received message is as follows:

    n text of message

where n is the number of the port which sent the message.

#### 3.14.4 Aborts and Recovery

Not applicable.

#### 3.14.5 Errors

An error in the entry of the MESSAGE command produces the display INVALID COMMAND.

### 3.14.6 Performance

Not applicable.

### 3.14.7 Installation Parameters

Not applicable.

## 3.15   JOB TERMINAL REQUEST

### 3.15.1 Abstract

In  order for communications between   a  job and a terminal to
begin,  a logical  connection or  linkage must be  established
between the terminal and the job. Normally for this linkage to
be established  action  is  required  by both the job  and the
terminal user.  The  actions required by  the job to establish
communications is described in the following paragraphs.


### 3.15.2 Description

The establishment of the job and   terminal linkage begins with
a terminal request by the job. The job uses the DEVICE control
card  or  Executive  Service  Request  (ESR)  to initaite  the
terminal request. The ability of the job to place restrictions
upon the port,  terminal,  and user is  also supplied  by  the
DEVICE control card or ESR.  Before the job regains control of
the CPU the terminal request parameters are validated.

If the job requested immediate connection to a port,  the  job
may begin communications with the terminal. If the job did not
request  immediate connection  to the port then communications
with the terminal  will be delayed  until the  user links with
this request.


### 3.15.3 Interfaces

For Requesting  a terminal the  format of  the DEVICE  control
card is:

         LU

    *DEVICE(lu=TERM,IC,~~LP=port,~~PP=port,VAL0=mask,VAL1=mask,
         CLASS=mask,UN=username)

                              P = ANY

where:

    lu is the logical unit.

    IC is the immediate connection flag.

    ~~LP=port,   port  is the logical port number~~.  This parameter
    may not be supplied if  the PP=port parameter is  supplied.
    If  the port  string  is an asterisk then  the submitting
    logical port is used.

                                               

PP=port, port is the physical port number, and this parameter string may not be supplied with the LP=port parameter.

VAL0=mask, the mask is a hexidecimal string where the linkage is restricted to ports and user which satisfy the following equation:

VAL0 mask .AND. validation mask = 0          *VAL0 = 1 = CAN'T USE THIS VALIDATION*

*(handwritten: VAL0 ... USER ... VAL0 = 1 = CAN'T USE THIS VALIDATION)*

VAL1=mask, the mask is a hexidecimal string where the linkage is restricted to ports and user which satisfy the following equation:

VAL1 mask .AND. .NOT. validation mask = 0          *VAL1 = 1 = MUST HAVE THIS VALIDATION*

*(handwritten: VAL1 ... USER ... VAL1 = 1 = MUST HAVE THIS VALIDATION)*

CLASS=mask, the mask is a hexidecimal string where the linkage is restricted to terminals which satisfy the following equation:

.NOT. mask .AND. 2**terminal class = 0

UN=username, if supplied the linkage to the terminal will be restricted to the specified username. To specify a group of usernames question marks are substituted for characters which need not match.

A request for a terminal using the DEVICE ESR has the following format:

|  |  | 0 | 7 8 | 1 1 5 6 | 2 2 3 4 | 3 1 |
|---|---|---|---|---|---|---|
| Reg n | | Address of Device Assignment Area | | | | |
| Reg n+1 | | logical unit | | | | |
| Reg n+2 | | Hardware Type 13 | | | | |
| Reg n+3 | | not used | | | | |

The format of the Device Assignment Area is as follows:

```
                              1 1          2 2          3
           8          7 8     5 6          3 4          1
         +----------+----------+----------+----------+
address  |                                          |
         +-----       Linkage Name       ------+
address+1|                                          |
         +----------+----------+----------+----------+
address+2| I A|          |          |     | Port    |
         +----------+----------+----------+----------+
address+3|          Validation Mask 0               |
         +----------+----------+----------+----------+
address+4|          Validation Mask 1               |
         +----------+----------+----------+----------+
address+5|                                          |
         +-------      User Name      ------+
address+6|                                          |
         +----------+----------+----------+----------+
address+7|          Terminal Class Mask             |
         +----------+----------+----------+----------+
```

*I = IMMEAD*
*A = Any Port*

### 3.15.4 Aborts and Recovery

The job is aborted if the device assignment area is in a fully protected page of memory or the logical unit is already assigned.

### 3.15.5 Errors

Not applicable.

*WHAT IS THE RETURN (IN PARM?) FROM THIS ESR?*

### 3.15.6 Performance

Not applicable.

### 3.15.7 Installation Parameters

Not applicable.

## 3.16   USER/JOB CONNECTION

### 3.16.1 Abstract

The terminal user may communicate with a process only after the job has requested a terminal. If the job did not request immediate connection to a port, then the user must use the LINK or KEEP command in conjunction with the linkage name to establish the process/terminal connection.

### 3.16.2 Description

The terminal user may have more than one processes connected to his port. One of these processes may be the current process, and it alone may communicate with the terminal. The LINK and KEEP commands are used to manipulate the connected processes.

The Link and KEEP command differ only in how the current process is handled. The LINK command disconnects the current process from the terminal, allowing no further communications between the terminal and that process to occur. The KEEP command, on the other hand, saves the connection to the current process for later terminal and process communications. Both the KEEP and LINK commands result in the specified linkage name process to become the current process, provided the specified process was a connected process or had been a process waiting for connection. If no process with the specified linkage name is available for connection or there is a validation conflict, then a diagnostic message is produced at the terminal.

A job may only become the current process of a port through one of the following sequences of events:

1.  The *DEVICE control statement is encountered with immediate connection requested.

2.  The job requests immediate connection to a port through the DEVICE ESR.

3.  The job requests connection to a port with the *DEVICE control statement or the DEVICE ESR, without requesting immediate connection, and the terminal user requests connection to the job using the LINK or KEEP command.

As previously described, the terminal user may disconnect from a process by using the LINK command to connect to another process.

The *CLOSE job control statement causes the specified logical units to be unlinked from the port. If no logical units remain linked, the job disconnects itself from the port.

### 3.16.3 Interfaces

The formats of the LINK and KEEP commands are:

    ]LINK PROCESSX
    ]KEEP PROCESSX

where PROCESSX represents the name of the process to which connection is desired. The process name of a job is the job id specified in the job's *JOB statement.

Successful connection is indicated by a message of the form:

    PROCESSX.

An unsuccessful connection attempt results in a message of the form:

    NO PROCESSX.

The format of the *CLOSE job control statement is:

    *CLOSE(lu,lu,...,lu)

where lu represents a logical unit number which is to be disconnected from the port.

### 3.16.4 Aborts and Recovery

Normal or abnormal job termination results in all logical units to be disconnected from the port.

Port recovery may only be performed on a port which had a connected process.

### 3.16.5 Errors

If the LINK and KEEP statements cannot be deciphered, the message INVALID COMMAND is produced.

If the desired process cannot be connected by the LINK or KEEP statements, the message NO PROCESSX. is produced where PROCESSX represents the requested process name. This results in the port having no current process.

### 3.16.6 Performance

Not applicable.

### 3.16.7 Installation Parameters

Not applicable.

## 3.17  TASK/TERMINAL COMMUNICATIONS

### 3.17.1 Abstract

After the terminal and job linkage has been established, a task may communicate with the terminal by using the MPX/OS Input/Output Subsystem. The valid MPX/OS I/O ESRs for a terminal are described in the follwoing paragraphs.

### 3.17.2 Description

Busy Status Executive Service Request (BSY)

The BSY ESR returns the busy/not busy status of the specified logical unit. The status is not a function of a particular I/O request but rather of the logical unit itself. The requesting task is scheduled after the request is completed.

Unit Status Executive Service Request (UST)

The UST ESR returns the status of the requesting task's last I/O request on the specified logical unit. If the I/O is still pending, the requesting task is placed in I/O wait until the I/O is completed.

Multiple Unit Status Executive Service Request (MUST)

The MUST ESR allows the requesting task to be placed in a wait state pending the occurance of the specified event(s). The requesting task supplies a 128 bit mask where bits 1-63 correspond with the end of operation on logical units 1-63 respectively. The other bit (0, 64-127) correspond to the user defined events (DEFEVNTQ ESR).

Unit Hardware Type Executive Service Request (UTYP)

The UTYP ESR returns the hardware type of the specified logical unit. ~~If the hardware type is a disk file, additional file description information is also returned.~~ The requesting task is scheduled for execution after the request is completed.

IN THIS CASE = IT ?

~~Standard files such as standard input, output, and are disk files not the card reader, or line printer.~~

## Status Logical Unit Executive Service Request (STATUS)

The STATUS ESR is called by a task to issue a status command to a specified logical unit. If the device is busy, the request is queued and the task is put into I/O wait. If the device is not busy, the current status is requested from the device and the task is put into I/O wait and not scheduled for execution until the operation is complete.

## Read Record from Logical Unit Executive Service Request (READLU)

The READLU ESR initiates a data transfer from a specified logical unit to a buffer residing in the requesting task's memory. The requesting task is scheduled for execution after the request is initiated, and must issue a BSY, UST, or HUST ESR to determine when the request is completed.

## Write Record to Logical Unit Executive Service Request (WRITLU)

The WRITLU ESR initiates a data transfer to a specified logical unit from a buffer residing in the requesting task's memory. The requesting task is scheduled for execution after the request is initiated, and must issue a BSY, UST, or HUST ESR to determine when the operation is completed.

## Clear a Logical Unit Executive Service Request (CLEAR)

The CLEAR ESR is called by a task to clear out the current I/O operation for a specified device. The command or data transfer is terminated at the point of receipt of the clear command. The task is put into I/O wait and not scheduled for execution until an end-of-operation interrupt is recevied. If the device is not busy, the clear request results in no action and the task is scheduled for immediate execution.

Select Operating Modes Executive Service Request (SELECT)

The SELECT ESR is used to set the operating mode for a logical unit. For terminals the SELECT ESR may be used to:

1. Request 96 or 64 ASCII character set on input.

2. Request line or block mode on input.

3. Request disabling of prompts.

4. Define prompt string.


Unsolicited Interrupt Request Executive Service Request (UINT)

The UINT ESR allows the requesting task to be notified when an unsolicited interrupt occurs on a logical unit. The unsolicited interrupt could be used to signal the occurance of a mal-function.


3.17.3 Interfaces

The BUSY ESR format for a terminal is as follows:

```
                          1 1          2 2           3
          0         7 8   5 6          3 4           1
          +------------+------------+------------+------------+
Reg n     |                   LU                            |
          +------------+------------+------------+------------+
Reg n+1   |                 NOT USED                        |
          +------------+------------+------------+------------+
Reg n+2   |                 NOT USED                        |
          +------------+------------+------------+------------+
Reg n+3   |                 NOT USED                        |
          +------------+------------+------------+------------+
```

LU          Logical unit for which busy status is requested.


An example of a calling sequence is as follows:

```
          EXT       BSY          Externally defined symbol
          LDI,R0    10           Logical unit number
          MON,R0    BSY          Monitor request
```

The parameters returned in PARM are as follows:

```
                          1 1          2 2          3
        0           7 8   5 6          3 4          1
        +-----------+-----------+-----------+-----------+
PARM    |                  BUSY FLAG                    |
        +-----------+-----------+-----------+-----------+
```

Busy Flag = Zero, unit not busy.
          = Non-zero, unit busy.


The UST ESR format for a terminal is as follows:

```
                          1 1          2 2          3
        0           7 8   5 6          3 4          1
        +-----------+-----------+-----------+-----------+
Reg n   |                    LU                         |
        +-----------+-----------+-----------+-----------+
Reg n+1 |                  NOT USED                     |
        +-----------+-----------+-----------+-----------+
Reg n+2 |                  NOT USED                     |
        +-----------+-----------+-----------+-----------+
Reg n+3 |                  NOT USED                     |
        +-----------+-----------+-----------+-----------+
```

LU          Logical unit for which status is to be returned.


An example of a calling sequence is as follows:

```
        EXT         UST         Externally defined symbol
        LDI,R0      10          Logical unit number
        MON,R0      UST         Monitor request
```

The parameters returned in PARM are as follows:

```
                          1 1          2 2          3
        0           7 8   5 6          3 4          1
        +-----------+-----------+-----------+-----------+
PARM    |                    US                         |
        +-----------+-----------+-----------+-----------+
PARM+1  |                    ES                         |
        +-----------+-----------+-----------+-----------+
```

US          Normal unit status
            bit 21 = 0.
                22 = 1, if BREAK detected.
                23 = 1, if terminal connection lost.
                24 = 0.
                25 = 0.

```
                        26 = 0.
                        27 = 0.
                        28 = 0.
                        29 = 0.
                        30 = 0.
                        31 = 0.
```

ES          Expanded status
            Number of bytes transfered.


The MUST ESR format for a terminal is as follows:

| | 0 | 7 8 | 1 1<br>5 6 | 2 2<br>3 4 | 3<br>1 |
|---|---|---|---|---|---|
| Reg n | | | Mask bits (0-31) | | |
| Reg n+1 | | | Mask bits (32-63) | | |
| Reg n+2 | | | Mask bits (64-95) | | |
| Reg n+3 | | | Mask bits (96-127) | | |


The parameters returned in PARM are as follows:

| | 0 | 7 8 | 1 1<br>5 6 | 2 2<br>3 4 | 3<br>1 |
|---|---|---|---|---|---|
| PARM | | | US | | |
| PARM+1 | | | ES | | |
| PARM+2 | | | Event bits (0-31) | | |
| PARM+3 | | | Event bits (32-63) | | |
| PARM+4 | | | Event bits (64-95) | | |
| PARM+5 | | | Event bits (96-127) | | |


US          Unit status as described for the UST ESR.

ES          Expanded status as described for the UST ESR.

The Event bits (0-127) are set to a one for each event which
has occurred.

An example of a calling sequence is as follows:

```
EXT        MUST         Externally defined symbol
LDD,R0     MASK1        Event Mask Bits
LDD,R2     MASK3        Event Mask Bits
MON,R0     MUST         Monitor Request
```

The UTYP ESR format for a terminal is as follows:

```
                                   1 1           2 2           3
               0           7 8     5 6           3 4           1
               +-------------+-------------+-------------+-------------+
Reg n      |                         LU                              |
               +-------------+-------------+-------------+-------------+
Reg n+1    |                       NOT USED                          |
               +-------------+-------------+-------------+-------------+
Reg n+2    |                       NOT USED                          |
               +-------------+-------------+-------------+-------------+
Reg n+3    |                       NOT USED                          |
               +-------------+-------------+-------------+-------------+
```

LU          Logical unit for which hardware type is requested.


An example of a calling sequence is as follows:

```
EXT        UTYP         Externally defined symbol
LDI,R0     10           Logical unit number
MON,R0     UTYP         Monitor request
```

The parameters returned in PARM are as follows:

```
                                   1 1           2 2           3
               0           7 8     5 6           3 4           1
               +-------------+-------------+-------------+-------------+
PARM       |                         HT                              |
               +-------------+-------------+-------------+-------------+
PARM+1     |                       not used                         |
               +-------------+-------------+-------------+-------------+
PARM+2     |                       not used                         |
               +-------------+-------------+-------------+-------------+
PARM+3     |                       not used                         |
               +-------------+-------------+-------------+-------------+
PARM+4     |                        Flags                           |
               +-------------+-------------+-------------+-------------+
```

HT          Hardware type
            = 14, terminal

```
Flags          Bit 0 = 0.
               Bit 1 = 0.
               Bit 2 = 0.
               Bit 3 = 0.
               Bit 4 = 1, if terminal is a receive-only terminal.
               Bit 5 = 1.
               Bit 6 = 1, if terminal is an interactive terminal.
               Bit 7 = 1, if terminal is a remote batch terminal.
```

The STATUS ESR format for a terminal is as follows:
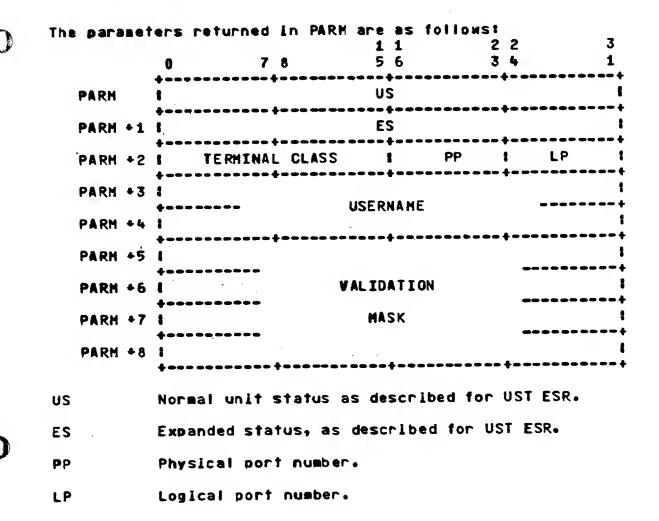
```
                             1 1          2 2              3
           0         7 8     5 6          3 4              1
           +-------------+-------------+-------------+-------------+
  Reg n    |                     LU                              |
           +-------------+-------------+-------------+-------------+
  Reg n+1  |                  NOT USED                           |
           +-------------+-------------+-------------+-------------+
  Reg n+2  |                  NOT USED                           |
           +-------------+-------------+-------------+-------------+
  Reg n+3  |                  NOT USED                           |
           +-------------+-------------+-------------+-------------+
```

LU             Logical unit


An example of a calling sequence is as follows:

```
        EXT       STATUS        Externally defined symbol
        LDI,R0    10            Logical unit number
        MON,R0    STATUS        Monitor request
```

The parameters returned in PARM are as follows:

```
                                   1 1           2 2           3
           0              7 8      5 6           3 4           1
         +-------------+------------+-------------+------------+
PARM     |                         US                         |
         +-------------+------------+-------------+------------+
PARM +1  |                         ES                         |
         +-------------+------------+-------------+------------+
PARM +2  |  TERMINAL CLASS         |     PP      |    LP      |
         +-------------+------------+-------------+------------+
PARM +3  |                                                    |
         +---------                 USERNAME         ---------+
PARM +4  |                                                    |
         +-------------+------------+-------------+------------+
PARM +5  |                                                    |
         +----------                                ---------+
PARM +6  |                      VALIDATION                    |
         +-----------                               ---------+
PARM +7  |                        MASK                        |
         +----------                                ---------+
PARM +8  |                                                    |
         +-------------+------------+-------------+------------+
```

US          Normal unit status as described for UST ESR.

ES          Expanded status, as described for UST ESR.

PP          Physical port number.

LP          Logical port number.

The READLU ESR format for a terminal is as follows:

```
                        1 1             2 2             3
        0         7 8   5 6             3 4             1
        +-----------+---------------+---------------+-----------+
Reg n   |                      ADDRESS                         |
        +-----------+---------------+---------------+-----------+
Reg n+1 |                      LENGTH                          |
        +-----------+---------------+---------------+-----------+
Reg n+2 |                      MODE                            |
        +-----------+---------------+---------------+-----------+
Reg n+3 |                      LU                              |
        +-----------+---------------+---------------+-----------+
```

ADDRESS    The address of the first element (word or byte) of
           the buffer is determined by the MODE parameter.

LENGTH     The number of words (bytes) to be transfered. LENGTH
           values may be from 0 to 4096 words (0 to 16,384
           bytes). A value of zero (0) is treated as the
           maximum LENGTH value of 4096 words (16,384 bytes).

MODE       The data Transmission Mode (format) code:

           0 ASCII record, word format
           16 ASCII record, byte format
           32 Binary record, word format

LU         Logical unit to be read.

An example of the calling sequence is as follows:

```
        EXT       READLU          Externally defined symbol
        LDCA,R0   BUFA            Byte address of buffer
        LDI,R1    48              Number of bytes
        LDI,R2    16              ASCII records in byte format
        LDI,R3    15              Logical unit number
        MON,R0    READLU          Monitor request
```

The WRITLU ESR format for a terminal is as follows:

```
                            1 1           2 2           3
            0        7 8    5 6           3 4           1
            +-------------+-------------+-------------+-------------+
Reg n       I                      ADDRESS                         I
            +-------------+-------------+-------------+-------------+
Reg n+1     I                      LENGTH                          I
            +-------------+-------------+-------------+-------------+
Reg n+2     I                      MODE                            I
            +-------------+-------------+-------------+-------------+
Reg n+3     I                      LU                              I
            +-------------+-------------+-------------+-------------+
```

ADDRESS   The address of the first element (word or byte) of
          the buffer is determined by the MODE parameter.

LENGTH    The number of words (bytes) to be transfered. LENGTH
          values may be from 0 to 4096 words (0 to 16,384
          bytes). A value of zero (0) is treated as the
          maximum LENGTH value of 4096 words (16,384 bytes).

MODE      The data Transmission Mode (format) code:

          0 ASCII record, word format
          16 ASCII record, byte format
          32 Binary record, word format

LU        Logical unit to be read.

An example of the calling sequence is as follows:

```
        EXT        WRITLU        Externally defined symbol
        LDCA,R0    BUFA          Byte address of buffer
        LDI,R1     48            Number of bytes
        LDI,R2     16            ASCII records in byte format
        LDI,R3     15            Logical unit number
        MON,R0     WRITLU        Monitor request
```

The CLEAR ESR format for a terminal is as follows:

```
                                1 1         2 2           3
          0           7 8       5 6         3 4           1
          +-------------+------------+-------------+------------+
Reg n     !                         LU                         !
          +-------------+------------+-------------+------------+
Reg n+1   !                      NOT USED                      !
          +-------------+------------+-------------+------------+
Reg n+2   !                      NOT USED                      !
          +-------------+------------+-------------+------------+
Reg n+3   !                      NOT USED                      !
          +-------------+------------+-------------+------------+
```

LU          Logical unit

An example of a calling sequence is as follows:

```
        EXT        CLEAR           Externally defined symbol
        LDI,R0     20              Logical unit number
        MON,R0     CLEAR           Monitor request
```

The parameters returned in PARM are as follows:

```
                                1 1         2 2           3
          0           7 8       5 6         3 4           1
          +-------------+------------+-------------+------------+
PARM      !     !C!                                            !
          +-------------+------------+-------------+------------+
```

C          Clear status, if set an operation in progress was
           cleared, otherwise the device was not busy.

The SELECT ESR format for a terminal is as follows:

```
                              1 1           2 2           3
          0           7 8     5 6           3 4           1
          +-------------+-------------+-------------+-----------+
Reg n     |                         LU                         |
          +-------------+-------------+-------------+-----------+
Reg n+1   |                        MODE                        |
          +-------------+-------------+-------------+-----------+
Reg n+2   |                      NOT USED                      |
          +-------------+-------------+-------------+-----------+
Reg n+3   |                      NOT USED                      |
          +-------------+-------------+-------------+-----------+
```

LU          Logical unit

MODE        bit 31 set; 64 ASCII character set requested.
                   clear; 96 ASCII character set requested.
               30 set; Block mode requested on input.
                   clear; Line mode requested on input.
               29 set; Disable prompting.
               28 set; next WRITLU defines prompt string.

An example of a calling sequence is as follows:

```
          EXT         SELECT          Externally defined symbol
          LDI,R0      12              Logical unit number
          LDI,R1      1               Select mode
          MON,R0      SELECT          Monitor request
```

The UINT ESR format for a terminal is as follows:

```
                              1 1           2 2           3
          0           7 8     5 6           3 4           1
          +-------------+-------------+-------------+-----------+
Reg n     |                         LU                         |
          +-------------+-------------+-------------+-----------+
Reg n+1   |                      not used                      |
          +-------------+-------------+-------------+-----------+
Reg n+2   |                      not used                      |
          +-------------+-------------+-------------+-----------+
Reg n+3   |                      not used                      |
          +-------------+-------------+-------------+-----------+
```

LU          Logical Unit assigned to the monitored device.

An example of the calling sequence is as follows:

```
EXT      UINT        Externally defined symbol
LDI,R0   15          Logical unit number
MON,R0   UINT        Monitor request
```

### 3.17.4 Aborts and Recovery

Errors made in an ESR may cause the job to be aborted, refer to the MPX/OS Reference Manual.

### 3.17.5 Errors

Errors made in an ESR are detected by the MPX/OS, refer to the MPX/OS Reference Manual.

### 3.17.6 Performance

Not applicable.

### 3.17.7 Installation Parameters

Not applicable.

## 3.18   INTERACTIVE JOB PROCESSING

### 3.18.1 Abstract

Interactive job processing allows the terminal user to enter job control statements from the terminal for immediate execution and to receive job standard output at the terminal.

### 3.18.2 Description

Interactive job processing provides for the entry of job control statements and the display of job standard output at the terminal. Whenever the standard input (logical unit 63) of the job is connected to the terminal, job control statements are read from the terminal; and whenever standard output (logical unit 62) of the job is connected to the terminal, job standard output is routed to the terminal. The connections of standard input and standard output to the terminal are independent.

Connection of standard input to the terminal allows the terminal user to directly enter job control statements to the Job Manager. Each statement is executed immediately after entry, and the appearance of the Job Manager prompt at the terminal indicates completion of the previous statement and the user may enter the next job control statement. The following job control statements may not be entered from the terminal and must be read from the cataloged job file before the standard input is connected to the terminal: *JOB, *RJOB, *IJOB, *IRJOB, and *SCHED. All other job control statements may be entered from the terminal in the interactive mode. The statement *CLOSE(63) causes job control statement input to be reconnected to the original input file, where the next statement to be read is the statement immediately following the *DEVICE statement which connected the job standard input to the terminal.

Connection of standard output to the terminal allows the terminal user to directly view job standard output. While this connection exists, no information is written on the job standard output file. All standard output from the job is displayed at the terminal with the following exceptions:

1. Job control statements are not output to the terminal.

2. Load maps are not output to the terminal.

The statement *CLOSE(62) causes job standard output to be reconnected to the original output file.

In the event of job termination, either normal or abnormal, all job logical units connected to the terminal are automatically disconnected and the job terminates as if no terminal had been connected. If the job standard output is connected to a terminal at the time of an abnormal job termination, an abort message is output to the terminal before the disconnection is performed.

### 3.18.3 Interfaces

All MPX/OS job control statements may be entered from the terminal with the following exceptions:

    *JOB
    *RJOB
    *IJOB
    *IRJOB
    *SCHED

These statements must come from the cataloged job file.

The Job Manager prompt appears as

    *?

and the user entry overwrites the prompt when the job control statement is entered.

The job abort message output to the terminal appears as

    JOB ABORTED ABORT TYPE = XX ABORT CODE = XX TASK = XXXX

### 3.18.4 Aborts and Recovery

In the event of a job abort, an abort message is output to the terminal and all logical units are disconnected from the terminal. Any resulting abort dumps are placed into the job standard output file.

Port recovery may be performed on ports which are connected to a job.

### 3.18.5 Errors

Errors are detected by the Job Manager, refer to the MPX/OS Reference Manual.

### 3.18.6 Performance

Interactive job processing affects system performance to the extent that the job requires system resources.

### 3.18.7 Installation Parameters

Not applicable.

### 3.19 PORT DISESTABLISHMENT

#### 3.19.1 Abstract

Port disestablishment is the disconnection of a terminal from the Interactive Terminal Subsystem. This disconnection may occur due to an explicit user command, failure to pass the validation check, or because of communications problems between the terminal and the system.

#### 3.19.2 Description

A logical port can be disestablished in three ways: by entry of the QUIT command, by entry error durring the validation check, or by loss of communications with the system.

At any time during the terminal session, the user may enter the QUIT command to terminate the session and disconnect the terminal from the Interactive Terminal Subsystem. A logoff message is output to the terminal to indicate the terminal disconnection. After this command is entered, any jobs in the system which attempt to communicate with this port are notified that the port has become disconnected.

Continued entry errors during the validation check has the same results as entering the QUIT command.

Loss of communication with the system results in either of two actions: if the port was not in communication with any jobs in the system, the effect is the same as entering the quit command. However, if the port was in communications with one or more jobs in the system, the port is placed in a recoverable condition. All port/job linkage information is retained and any outstanding and subsequent input/output requests are saved for future recovery of the port. The recovery information is retained within the system for a predefine period of time. When this retention period has elapsed, if the port has not been recovered, all processes trying to comunicate with the port are notified that the port has become disconnected. Detection of a communications lose is dependent upon specific site terminal input/output implementation.

### 3.19.3 Interfaces

The format of the QUIT command is:

    ]QUIT

Whenever the terminal is diconnected from the system, the following message containing the current time of day and total connect time is produced:

    LOGOFF AT 4:32:53
    CONNECT TIME 98.03 SECONDS.

### 3.19.4 Aborts and Recovery

If the port is connected to a job when communications are lost between the terminal and the system, then the port will be placed in a recoverable condition. The Port Recovery feature describes the method of recovering the port. This feature does not participate in aborts.

### 3.19.5 Errors

If the QUIT command is entered incorrectly, the message INVALID COMMAND is displayed.

### 3.19.6 Performance

Not applicable.

### 3.19.7 Installation Parameters

The port recovery time-out is an installation parameter.

## 3.20  PORT RECOVERY

### 3.20.1 Abstract

The terminal user may recover a previously interrupted terminal session.

### 3.20.2 Description

After establishing the terminal as a logical port within the Interactive Terminal Subsystem, the user may enter the RECOVER command to attempt recovery of a previously interrupted terminal session. The user must not have any connected processes when the recovery is attempted and the user's username must match the username of the port to be recovered. The user can recover an interrupted terminal session from any terminal, therefore the user must specify the number of the logical port to be recovered.

Recovery is limited to ports which were in communications with jobs in the MPX/OS system at the time of communications loss, and the port recovery must be initiated within a limited time interval (see Port Disestablishment).

### 3.20.3 Interfaces

The format of the RECOVER command is:

    )RECOVER n

where n represents the number of the port to be recovered.

A typical port recovery scenario proceeds as follows:

    Recovery Command:     )RECOVER 2
    Response:             RECOVERY COMPLETE.

At this point, the user has regained control of the interrupted terminal session. However, since recovery of a terminal session may be made from any terminal, the baud rate, parity, and duplex are not recoverable. Input mode and character set are recovered.

### 3.20.4 Aborts and Recovery

The port recovery feature provides the only means of recovering an interrupted terminal session. In the event of a system abort, all terminal sessions are lost and cannot be recovered.

### 3.20.5 Errors

If the RECOVER command is entered incorrectly, the INVALID COMMAND message is produced.

If the port can not be recovered the message RECOVERY NOT POSSIBLE is produced. This may be caused by incorrect entry of the port number, by attempted recovery with a different username, or because the port is not in a recoverable condition.

### 3.20.6 Performance

Not applicable.

### 3.20.7 Installation Parameters

The recovery time out value is an installation parameter, changable by recreating the system resident.

### 3.21   TERMINAL USER ASSISTANCE

### 3.21.1 Abstract

The terminal user may get on-line assistance in the use of ITS features.

### 3.21.2 Description

The HELP command allows users to get immediate assistance from the Interactive Terminal Subsystem on using ITS commands and features. The HELP command displays commands their parameters and options and a description of what the command does.

### 3.21.3 Interfaces

The format of the HELP command is:

    ]HELP cmd

where cmd represents the command the user wants information on. If no command is specified a list of all ITS commands with a brelf description about each is displayed.

### 3.21.4 Aborts and Recovery

Not applicable.

### 3.21.5 Errors

If the command specified is not a valid ITS command the message NO SUCH COMMAND is produced.

If the help command is not recognizable the message INVALID COMMAND is produced.

### 3.21.6 Performance

Not applicable.

### 3.21.7 Installation Parameters

Not applicable.

**O**

## 4.0   PRODUCT-LEVEL DESCRIPTION

## 4.1   PUBLICATIONS AFFECTED

14306500     MP-60 Computer System Family Reference Manual

18817300     MP-60 Computer System MPX/OS Reference Manual

10992100     MP-60 Computer System MPX/OS Operator's Guide

11110000     MP-60 Computer System Installation Handbook

14291700     MP-60 Computer System Program Command Console
             Reference Manual

             MP-60 Computer System Text Editor Reference Manual


## 4.2   EQUIPMENT CONFIGURATION

Equipment configurations for the MPX/OS system with the
Interactive Terminal Subsystem are as follows:

**O**

|                          | MINIMUM | TARGET | MAXIMUM |
|--------------------------|---------|--------|---------|
| Number of CPUs           | 1       | 1      | 8       |
| Program states           | 6       | 6      | 48      |
| Memory (32-bit words)    | 98,304  | 131,072| 4,194,304 |
| Card reader              | 1       | 1      | 1       |
| Line printer             | 1       | 1      | 2       |
| Display console          | 1       | 1      | 1       |
| Interactive terminals    | 0       | 48     | 256     |
| Mass storage (megabytes) | 10      | 400    | 1500    |
| Magnetic tapes           | 0       | 2      | 8       |

**O**

## 4.3    INTERFACES TO OTHER SOFTWARE PRODUCTS

The amount of memory memory available for user job allocation is divided into two pools during system initialization. The rollin pool is used for the initiation and execution of jobs beginning with the job control statements *IJOB and *IRJOB. The remaining memory pool is available to the system for normal usage.

Rollin/Rollout file space is allocated during system initialization. This file space receives the pages of rolled out jobs at a 1 page per block usage rate.

Jobs may be rolled out when they are waiting for the receipt of buffered messages. They are rolled in upon the receipt of a message or upon system operator termination.

Whenever a job is executing, its full amount of scheduled memory is available for expansion. Only those pages actually in use are rolled out. The pages of a job are rolled out as rollin requirements demand additional space: the system attempts to maintain enough free space to rollin the average sized job.

In the event that a task sends a message to a job which is in a rollout state, the system processes the message if the recipient buffer pages are resident, otherwise the sending task is placed in a rollin wait state until the recipient has been returned to memory.

The Task Control Table contains the rollout file location of each page which is not memory resident.


## 4.4    RESTRICTIONS AND LIMITATIONS

Detection of a communications loss with a terminal and control of transmission formats between the terminal and the system are dependent upon specific site implementation. Therefore, the port recovery and port setup functions may not be available in all implementations.

Line editing, prompting, and terminal display formats have been chosen to be compatible with most terminal hardware types. Currently available Control Data terminal types 751, 752, and 756 are among the compatible terminals.

The introduction of interactive processing to any system greatly increases mass storage requirements as users place information into mass storage files which was previously maintained in card, listing, and magnetic tape media.

The execution of lengthy compute bound tasks should be restricted to the normal batch environment. Execution of these tasks in rollin pool memory will significantly degrade the system performance observed by terminal users.

4.5    RELIABILITY, AVAILABILITY AND MAINTAINABILITY REQUIREMENTS

Upon installation, the Interactive Terminal Subsystem becomes an integral part of the MPX/OS system, and therefore meets the reliability, availability, and maintainability requirements of the MPX/OS system.

# 5.0 GLOSSARY

Abort                The premature termination of a process
                     whenever an irrecoverable situation
                     occurs.

Absolute             Refers to actual machine address.

ASCII                American Standard Code for Information
                     Interchange. The standard code, using a
                     coded character set consisting of 7-bit
                     coded characters. See American National
                     Standard X3.4-1977.

Assemble             The process by which an object (binary)
                     module is created from the symbolic
                     language program.

Asynchronous         Refers to a type of serial transmission in
                     which bit synchronization is accomplished
                     for each character.

Batch                Class of tasks which run on a time
                     available basis.

BCLA                 Buffered Communications Line Adapter is
                     the controller interface to the RS-232C
                     asynchronous communications lines.

Bit                  A binary digit.

Block                A group of machine words or bytes. Usually
                     a collection of one or more records used
                     in I/O to reduce the number of physical
                     operations.

Buffer               A portion of memory used to collect data
                     in order to compensate for speed
                     differences between the processor and
                     peripheral devices.

Byte                 Eight (8) contiguous bits.

CALL                 The tranfer of control to a closed routine
                     or task. A Executive Service Request,
                     CALL, is used to activate a specific task.

Callee                  The task initiated by the CALL Executive
                        Service Request.

Caller                  The task initiating another routine or
                        task.

CLA                     Communication Line Adapter, the hardware
                        interface between the CPU and the
                        communications line.

Compile                 The process by which an assembly module is
                        created from a problem solving language. A
                        compiler usually generates several machine
                        instructions from a single symbolic
                        statement.

Common                  An area of memory that may be shared
                        between tasks or routines. Tasks may
                        communicate through common areas.

Control Point Zero      CYBER System Control Point.

CPU                     Central Processing Unit.

CRC                     Cyclic Redundancy Check.

Data                    An area of memory that may be prestored
                        with information at load time and may be
                        shared between routines of a task.

Dispatcher              An operating system routine that unthreads
                        a task from the top of the ready list and
                        places the task into execution.

ESR                     Executive Service Request, name given to
                        the MPX Operating System routines invoked
                        by the MON instruction.

File                    A collection of blocks and/or records,
                        usually of related data. Each mass storage
                        file has an entry in the system Label
                        File.

Interrupt               A break in the normal processing flow
                        usually caused by a hardware signal.
                        Interrupts can be enabled or disabled and
                        occur with an associated priority.
                        Processes that are interrupted are later

|  |  |
|---|---|
|  | resumed at the point of interruption. |
| ITS | Interactive Terminal Subsystem. |
| IOC | Input/Output Controller. |
| Job | The sequential and/or parallel execution of tasks. Begins with a *JOB card and ends with a *EOJ card. |
| JCT | Job Control Table is an area of memory containing information controlling a given job. |
| Job Manager | A system task that processes the input stream of the job. The Job Manager is a set of reentrant routines shared by all user jobs. |
| Library | A collection of frequently used, checked-out programs maintained on an external device that can be loaded and executed separately or in conjuction with a user's program. Libraries must be arranged to minimize searching. |
| Linkage | The interconnection between routines or devices. The loader matches externals and entry points to establish program linkage. The Interactive Terminal Subsystem match job terminal requests with terminal user connection requests to establish port linkage. |
| Loader | A system task that is used to load, relocate, and link binary object modules. |
| LU | Logical Unit, a number representing the user connection to a device. |
| MPX/OS | The MPX Operating System. |
| MP-32 | MP-60 CPU Emulation device. |

MPP      MP-60 CPU Emulation device (Militarized, or Ruggidized).

Ordinal      The relative location of an entry in a table. The absolute location of an entry can be obtained by multiplying the ordinal by the number of machine words per entry and adding the starting address of the table.

Page      A 4096 word block of contiguous memory. Paging is a technique where a logical address is transformed via a set of page registers into a physical address.

PMAC      Programmable Multiple Access Controller.

Po~t      The communications line between the CPU and the user.

PPJ      CDC CYBER/6000 Peripheral Processor Unit.

Process      A software program equivalent to the MPX/OS terms tasks and job.

Priority      A value assigned to an item in the system which facilitates scheduling and processing within the operating system.

Queue      A list used to control the processing to be done.

Ready list      A prioritized list of tasks waiting for control of the CPU.

Relocatable      Refers to a program that has been prepared by a source language compiler or assembler to be loaded into any area of available memory.

Resident      The portion of the operating system which resides permanently in memory.

RETURN      A Executive Service Request that terminates a task and transfers control to the point in the caller where the call originated. A task may return with or without release of memory.

| | | |
|---|---|---|
| Schedule | The determination of which processing is required next. | |
| Stack | A last-in, first-out queue. | |
| Status | A stage or condition of an I/O request or a task. | |
| Synchronous | Serial transmission in which characters are sent bitwise without start and stop bits. | |
| System Initialize | Refers to the intial system load process where the resident is loaded, memory initialized, and the system tasks are started. | |
| Task | An independent unit of work that can compete for the resources of the system. A task may call and be called by other tasks. | |
| TCT | Task Control Table is an area of memory containing information used to control a task. | |
| Terminal | The device connected to the user end of the PORT. | |
| Terminate | The process of completing a job. A job may terminate normally or abnormally. | |
| Thread | A linked list of elements, the contents of each thread cell contains the address of the next thread cell and so on until a thread cell contains a value indicating the end of the list. | |
| Utility | A routine, procedure or program that supports the operation of a system. | |

APPENDIX A

The following pages contain examples of the various Facility displays.

The Device Control Facility LIST command provides information about the status of peripheral devices. The IEU column provides the IOC, Equipment and Unit Numbers. The OWNER collumn serves two purposes. When an entry is present it indicates the device is assigned and whether it is a system or user device. When it is blank it indicates that the device is available to a user but is not currently assigned.

```
VPX-OS   RES R1   LIB L1      08.00.05    06/15/79   MEMORY AVAIL 10      USED 25

IEU      TYPE     OWNER    STATUS   ID

000      85PD     SYS      UP       SYSTEM01
001      85P      USER     UP       TESTPACK
101      CR       SYS      UP
102      LP       SYS      UP
200      MT7               DOWN
300      MT9      USER     UP
```

Fig. A-1   DCF LIST DISPLAY

The File Control Facility LIST command provides the terminal user with status information about mass storage files.

The "O" and "A" displays provide information about files under a specified owner on a particular SMD which are either open or allocated, respectively.

```
LBN  OWNER   FILE NAME      ED   ACC    TYPE   DEVICE ID
211  MINE    AFILE          01   MINE   858D   SYSTEM01
285  MINE    BLD-NEW-FILE   01   MINE   858D   SYSTEM01
286  MINE    OLD-FILE       01   MINE   858D   SYSTEM01
325  MINE    OLD-FILE       02   MINE   858D   SYSTEM01
```

The "O" display provides a list of the current SMD labels.

```
DEVICE ID
SYSTEM01
SYSTEM02
```

Fig. A-2  FCF LIST DISPLAYS

The Operator Control Facility LIST command provides several possible displays. The Input, Output, and Execution Queues may either be displayed individually or they may be displayed together.

The "I" display will provide expanded information about the INPUT queue.

```
VPX-OS  RES R1  LIB L1      08.00.20   06/15/79  MEMORY AVAIL 10      USED 25

NXTJOB    CM=15,TL=70000,PL=10000,SCR=100,MT
BNCHMRK   CM=20,TL=99999,PL=200,SCR=10,TESTPACK
QUIKONE   CM=5,TL=100,PL=100
```

The "M" display provides a list of the messages requiring operator response. Each entry has a unique message identifier which will be used to respond to the message. The identifier will be assigned to the message and will remain with the message until it receives some operator action.

```
01 08.00.04 YOURJOB   PAUSE CONTROL CARD MESSAGE APPEARS HERE
02 08.00.07 MYJOB     EQUIP
```

If just LIST is entered, all queues will be displayed.

```
VPX-OS  RES R1  LIB L1      08.00.15    06/15/79   MEMORY AVAIL 10      USED 25

INPUT            OUTPUT        DESTIN      EXECUTION

01 NEXTJOB       01 NEXTOUT    CENTRL      01 MYJOB
02 BNCHMRK       02 SYSLIST    REMOTE      02 YOURJOB
03 QUIKONE       03 TESTJOB    DISK
                 04 JOBTWO     CENTRL

01 08.00.04 YOURJOB   PAUSE CONTROL CARD MESSAGE APPEARS HERE
02 08.00.07 MYJOB     EQUIP
```

Fig. A-3   OCF LIST DISPLAYS

The Operator Control Facility WATCH command provides information about active jobs and their associated tasks.

The WATCH command with no parameters displays general information about all active jobs.

```
VPX-OS  RES R1  LIB L1       08.00.07     06/15/79    MEMORY AVAIL 10  USED 25

YOURJOB     MEM  SCHED 15  USED 07  **  TIME  SCHED INFINITE    USED 02.56.53
0D5E          ORIGIN 0000  IN DISP SC  PL 1000  PL DISP PR  PC 00000 PC DISP PU

VYJOB       MEM  SCHED 10  USED 06  **  TIME  SCHED 00.30.21    USED 00.04.07
0D70          ORIGIN 0000  IN DISP SC  PL 10000 PL DISP PR  PC 00000 PC DISP PU
```

If a "jobid" is specified, information about that particular job and all its associated tasks is displayed.

```
VPX-OS  RES R1  LIB L1       08.00.07     06/15/79    MEMORY AVAIL 10  USED 25

YOURJOB     MEM  SCHED 15  USED 07  **  TIME  SCHED INFINITE    USED 02.56.53
   1   JMTR      STATUS 000    P 0275     CPU 1      STATE 06  PRIORTY 000
   2   USERTASK STATUS 001    P D041     CPU 1      STATE 03  PRIORTY 000
```

If "jobid,taskid" is entered, information about that particular job and task is displayed.

```
VPX-OS  RES R1  LIB L1       08.00.08     06/15/79    MEMORY AVAIL 10  USED 25

YOURJOB     MEM  SCHED 10  USED 06  **  TIME  SCHED INFINITE    USED 02.56.53
0D5E          ORIGIN 0000  IN DISP SC  PL 1000  PL DISP PR  PC 00000 PC DISP PU
   2   USERTASK STATUS 001    P D041     CPU 1      STATE 03  PRIORTY 000
       0FD6      IN CNT 001    FLAGS 8100             TIME 00.03.00
                 EVENT BITS       00000000 00000000 00000000 00000000
                 EVENT WAIT BITS 00000000 00000000 00000000 00000000
```

Fig. A-4   OCF WATCH DISPLAYS

The Operator Control Facility MEMORY command provides the capability to display and alter memory.

The "S" display provides a Hexadecimal dump of those memory locations altered by a OCF MEMORY command. Up to a full screen will be displayed.

```
PPX-OS  RES R1  LIB L1       0F.00.10    06/15/79   MEMORY AVAIL 10    USED 25

20EE   00000000 00000000 00000000 00000000 00000000.00000000 00000000 00000000
```

The other displays provided through the OCF MEMORY command provide a Hexadecimal dump of memory starting at a specified location and ending at either a full screen or once the requested number of locations to be dumped has been displayed if less than a full screen.

```
PPX-OS  RES R1  LIB L1       0F.00.30    06/15/79   MEMORY AVAIL 10    USED 25

20EP   00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
20F0   BCC03EEA 2650003F BC94003F BC940000 BCC020B4 0EDE0011 C016B00E C00E7006
20F8   C0143004 0AB0004C E05507BF 72B5000E 22B53463 22B53463 27150007 13200000
2100   BA960001 1320FFFF 47370003 23970003 0FB30013 13C00001 BD000BAE BD000A3E
2108   13800000 BD3F3E6A BD000BAE BCC03E6A 72B50005 22B5355E 27150001 13200000
```

Fig. A-5  OCF MEMORY DISPLAYS